Rendimientos y mitos de los discos duros.

Uno de los mejores indicadores de rendimiento es el precio.

La velocidad de un disco se mide de dos formas:

- promedio de transferencia
- tiempo medio de acceso

Promedio de transferencia

Es probablemente el dato más importante para saber el rendimiento de un disco, pero debido a que se proporcionan varios valores (hasta siete) suele ser un dato que resulta confuso.

La velocidad media de transferencia es inferior a la de la de interfaz (en caso de SATA 150, 300, 600), por lo tanto no hay que dejarse engañar creyendo que la velocidad del disco viene determinada por la interfaz. La transferencia media hace referencia a la velocidad a la que el disco lee o escribe datos, y la de la interfaz indica cómo de rápido se mueven los datos entre la placa base y el buffer del disco.

Los promedios de transferencia tienen un máximo y un mínimo debido al ZBR. Los promedios raw hacen referencia a lo rápido que se lee de disco, pero incluye todos los bits (intersector, ECC, ID); en cambio los formatted representan el verdadero promedio de velocidad, ya que trabaja con los datos de usuario.

Algunos fabricantes sólo proporcionan los promedios de transferencia raw, pero se puede calcular los los promedios de datos formatted porque suelen ser 3/4 partes de los raw.

Los dos factores principales que determinan el promedio de transferencia son la velocidad de rotación y la densidad (sectores por pista). En el caso de dos discos con igual velocidad, la transferencia es mayor en el que tiene mayor velocidad de rotación; y en caso de igual densidad, la mayor transferencia la tiene el de mayor densidad. Pero es necesario tener en cuenta que un disco de mayor densidad puede ser más rápido que otro que gire a mayor velocidad.

Para ser exactos, hay un pequeño beneficio en interfaces de transferencia más rápida. Los datos del buffer de disco se transfieren a la placa base a la velocidad de la interfaz, en vez de la velocidad promedia.

Tiempo de búsqueda medio

Cantidad de tiempo promedio en ms que toma mover las cabezas de un cilindro a otro a una distancia aleatoria. Es un indicador claro de las capacidades del mecanismo accionador de la cabeza.

Latencia

Tiempo medio que toma a una cabeza situarse en un determinado sector de una pista. Es un factor en el rendimiento de lectura y escritura de un disco, ya que decrementando la latencia se incrementa la velocidad de acceso a los datos y se consigue aumentando la velocidad a la que giran los platos.

Incrementando la velocidad de los discos se incrementan el promedio de transferencia de datos después de que las cabezas lleguen a los sectores deseados.

Tiempo medio de acceso

Es la suma de la velocidad de acceso y la latencia, y se expresa en ms. Proporciona la cantidad total de tiempo promedio que se requiere para acceder a un sector aleatorio.

Cache

Antiguamente se utilizaban programas de caché de disco que ponen un hook en la interrupción de disco de la BIOS e intercepta las lecturas y escrituras. La mayoría de las controladoras actuales tienen algún caché en hardware que no intercepta ninguna interrupción de la BIOS. Una caché no afecta al verdadero promedio de transferencia que un disco admite.

Resumen

El promedio de acceso viene determinado por la densidad y velocidad de rotación. Y el tiempo de acceso por el mecanismo accionador de las cabezas y la velocidad de rotación.

La tasa viene determinada por la densidad, y no, la limitación no suele estar en la interfaz (otra cosa normalmente mal entendida). Las interfaces en SATA, por ejemplo, van de 150 MB (SATA I) a 300 (SATA II) y finalmente 600 MB/s (SATA III). Un disco duro más o menos estándar suele tener una transferencia de 90-100 MB/s* y eso usando la caché, así que el cuello de botella está en la tasa de transferencia, no en la interfaz. Usease, que en un SATA III ni te cuento.

* por supuesto no es una regla de oro, depende de discos.

Información respecto a una de las confusiones habituales: *IDE - ATA*

Normalmente IDE se utiliza para indicar la interfaz que conecta el disco duro al ordenador, pero el nombre correcto es ATA.

IDE

IDE hace referencia a la combinación disco/controladora, el cual hace uso de una conexión ATA. Por tanto, se podría decir que ATA es un subconjunto o versión específica de IDE. La tecnología IDE tiene varias ventajas:

- -mayor simplicidad al no ser necesaria una alimentación separada ni cables que conecten ambos.
- -las señales tienen que recorrer una menor distancia, por lo tanto son menos propensas al ruido y se incrementa la fiabilidad
- -incremento del promedio de reloj en el codificador y densidad del disco.

Los primeros discos IDE fueron los llamados hardcards, los cuales eran tarjetas que se pinchaban en la placa base. Tenían como inconveniente que pesaban bastante, y producían interferencias con el resto de tarjetas conectada.

ATA

Originalmente ATA era una interfaz paralela de 16 bits con un pinout de 40 pines, aunque posteriormente surgió una interfaz serie. Como el término ATA no implica necesariamente ninguno de estos dos tipos, hay que hacer referencia expresa:

- -SATA: serie
- -PATA: paralelo

SATA fue una interfaz serie introducida en 2000, que se empezó a utilizar en los sistemas de escritorio en 2003 y en los portátiles en 2005. Al ser serie sólo se envían un bit en un determinado momento de tiempo, por lo tanto el cable utilizado para la transmisión de datos es de menor tamaño. Al contrario de lo que se pueda pensar, proporciona un mayor rendimiento.

ATA y SATA son dos interfaces físicas totalmente diferentes, pero SATA mantiene compatibilidad a nivel de software con ATA paralelo.

Existen interfaces anteriores a ATA:

- -ST-506
- -ESDI

En ambos casos la controladora y el disco están separados.

En resumen: que IDE hace referencia a que la controladora está integrada en el disco, y ATA a la interfaz.

Mucha gente confunde el término controladora y *HBA*, este texto es para aclarar más el tema.

Según la cultura popular la controladora está en la placa base, es decir, externa al dispositivo. Como ya se ha visto es un error, porque justamente el término **IDE** viene a indicar que la controladora se incluye dentro del dispositivo.

La controladora se encarga de la parte lógica: el rotor, el cabezal, las L/E, la caché, ... Puede estar totalmente implementada por hardware, pero es una opción cara. Por eso se suele utilizar parte software: el *firmware*.

El **HBA** (*Host Bus Adapter*) se encarga de transferir datos entre el bus del dispositivo y el bus del ordenador. Viene a ser una especie de pasarela. Por ejemplo, con un dispositivo *SCSI* el *HBA* hace de intermediario entre el bus *SCSI* y el de la máquina.

Es importante darse cuenta de que lo que normalmente la gente llama controladora es un *HBA*. En la Wikipedia se recalca que hay un gran confusión entre estos dos conceptos.

Por tanto, lo que normalmente está integrado en la placa base es el *HBA*. Al menos para dispositivos *ATA* y *SATA*, que son utilizados en máquinas personales. En el caso de dispositivos *SCSI* sería necesario un HBA *externo*, por ejemplo, en forma de tarjeta *PCI*; ya que las placas bases comunes no lo suelen integrar. Por lo tanto, el *HBA PCI-SCSI* sólo es necesario si la placa base no soporta *SCSI* de forma nativa. Suelen ser caras.

La verdad estricta es que el *HBA* es el punto de unión de un bus con la placa base. Está en la propia placa base, pero interconecta el bus SCSI (por ejemplo) y el bus de la placa base.

Existen modelos que tienen una batería que alimenta una memoria para no perder transacciones si se va la luz. También algunos tienen la posibilidad tanto de conectar dispositivos internos, como ranuras para conexiones externas.

Es importante tener en cuenta que el *HBA* cuenta como uno de los dispositivos que se pueden conectar al cableado *SCSI*. Por tanto, en caso de soportar 16 dispositivos, realmente quedarían 15 libres.

En el caso de querer hacer **RAID** por hardware haría falta una controladora *RAID*. Como no suele estar presente en las placas bases (al igual que *SCSI*) haría falta una tarjeta externa *PCI*. En el caso de la tarjeta, sería tanto un *HBA* como una controladora, pero de *RAID*. Sólo las placas de gama media-alta tienen *RAID* por hardware.

Normalmente los dispositivos de almacenamiento para usuarios permiten *RAID* por software mediante una BIOS en el dispositivo. Es decir, BIOS en el dispositivo más controladora estándar; de forma que toda la gestión del *RAID* se realiza por software.

Por último hablo de un poco de la caché del disco duro y de la caché que implementa una controladora *RAID*. Todo lo que escribo es de los apuntes que me redacté a partir de una conversación con **Jcea**. Desde aquí le doy las gracias porque me ayudó a entenderlo.

FLUSH CACHE

Hace referencia a un comando de escritura en el cual se indica que se tiene que grabar directamente en disco y no en caché.

Existen dos usos típicos de *flush cache*:

1) -asegurar el orden de algunos cambios en el disco duro. Lo implementan los sistemas de ficheros, aunque no todos.
2) -obedecer directivas SYNC (llamadas al sistema) del sistema operativo

Los discos ante este tipo de escrituras pueden actuar correctamente, de forma que te dan un ACK cuando se ha escrito en disco. La otra posibilidad es devolverte un ACK habiendo escrito los datos en la caché y no en disco; en este segundo caso su respuesta no es veraz.

Se puede comprobar si un disco duro funciona correctamente con *flush cache* haciendo un programa que escriba datos al disco y haga un sync. Si el disco ha escrito muy rápido la información, es bastante probable que realmente no realice las escrituras sin pasar por la caché.

Si se realiza una escritura *flush cache* en una controladora RAID, ésta devuelve inmediatamente al sistema de ficheros un ACK como que se ha escrito con seguridad en el disco. A su vez envía comandos *flush cache* al disco que correspondan, pero guarda los datos en su caché hasta que confirmen su escritura mediante ACK (tiempo de ms). Si no se confirma la escritura y se va la luz, la batería previene la pérdida de datos de la caché de la controladora. Cuando hay corriente de nuevo, entonces la controladora volverá a mandar los comandos de escritura al disco correspondiente.

El uso de *flush cache* depende del driver y sistema de ficheros. ZFS, por ejemplo, hace uso del mismo.

NOTA: **Jcea** apunta que no sabe con seguridad si una escritura *flush cache* es un comando específico de escritura, o uno normal donde se indica mediante un bit. Piensa que es un comando separado.

CACHÉ DISCO DURO

No siempre hay que desactivar la caché para estar seguro de que los datos se escriben correctamente en disco. Si se cumplen dos condiciones:

1) -el disco duro escribe directamente ante *flush cache*
2) -el sistema de ficheros utiliza *flush cache* correctamente

La caché de escritura no es un problema, sino una gran ventaja.

Aunque mis conocimientos sobre write barriers son mínimos, pego unos pequeños apuntes que tengo sobre el tema.

Write Barriers: mecanismo del núcleo para asegurar que los metadatos del sistema de ficheros se escriben correctamente.

Para que el journaling funcione de forma consistente, primero necesita escribir en disco el cuerpo de la transacción y luego el commit block; en caso de perder la corriente el sistema de ficheros puede recuperar la transacción ante un fallo de alimentación. Pero es necesario que las escrituras se realicen en ese orden estricto, cosa que no sucede si el disco tiene una caché activa, pues el disco reordena las escrituras a su gusto. Para evitar eso se utiliza *barriers*, donde se fuerza mediante flush cache a que se escriba en orden (primero el cuerpo y luego el commit block de la transacción).

Existen dos casos en que no es necesario activar *write barriers* (mount -o nobarrier):

-dispositivos (controladoras RAID, por ejemplo) con batería para la caché de escritura*
-caché de escritura desactivada.

Es importante tener en cuenta que *write barriers* proporcionan consistencia de datos pero a costa de un peor rendimiento de escritura.

* en este caso los discos tienen que tenen la caché desactivada, o funcionar correctamente ante 'flush cache'.